FSA Modernization Partner

United States Department of Education

Federal Student Aid

Integrated Technical Architecture
Web Graphical User Interface (GUI) Standards

Task Order # 69

Deliverable # 69.1.4

Version 2.0

June 27, 2002

Table of Contents

1 Executive Summary

To ease navigation and promote consistent common look and feel among all web applications, the Web Graphical User Interface (GUI) Standards must be used by the web designers and web developers to ensure that all content within a web site, and to a certain degree, among all FSA web sites, should share the same writing style, graphics, colors, and user interface elements.

These guidelines present the overall FSA web philosophy and should facilitate automation of web site and page design. The guidelines promote completeness and a consistent interface among all FSA Web sites, but they also allow developers a certain degree of freedom and creativity, and the ability to tailor web applications according to its primary audience.

This is a working document which combines the best of many similar HTML guides (see References). Feedback and input to the GUI standards is requested and appreciated from all interested parties, especially FSA web designers and web developers.

This document includes:

- Guidelines for the creation of web content and multimedia files.
- HTML coding standards and usability standards.
- Examples of screen layout, navigation, headers, footers, and other design issues are provided in the form of generalized templates, which should be used as a model for developing and maintaining FSA web pages.
- The Additional Resources section points to other Style Guides, HTML specifications, HTML primers, quick references, and other helpful resources.

All FSA web sites and web applications should adhere to FSA web GUI Standards.

2 User Interface Considerations

A user-friendly interface makes it easy for users to navigate, for the computer to request information from the users, and for the computer to present understandable information. Clear communication between the user and the computer is the working premise of good UI design.

The user interface developers create should accomplish the following:

- The user interface must be visually appealing to users
- The user interface must provide easy access to information so first-time users don't become intimidated
- The interface and the information should reinforce the individual's understanding of the FSA community and their role as a member of it
- The information provided should be accurate and up to date -- where relevant
- The interface should keep the users and their goals in mind
- The interface should consider different levels of users experience (novice, experienced), user information and transaction needs and frequency of use

Good web sites generally embody the well-established design and user interface principles identified in Elements of Good User Interfaces (see Table 1 below).

Table 1: Elements of Good User Interfaces

User Interface Requirement	Description
Clear	A clear understandable and intuitive user interface helps prevent user errors, makes important information obvious, and contributes to ease of learning and use.
Simple	The best interface designs are simple. Simple designs are easy to learn and to use and give the interface a consistent look. A good design requires a good balance between maximizing functionality and maintaining simplicity through progressive disclosure of information, logical layout and consistent framework.
Aesthetic	Every visual element that appears on the screen potentially competes for the user's attention. Provide an environment that is pleasant to work with and contributes to the user's understanding of the information presented. Give prominence to the most relevant info or content on the page according to user need and usage.
User-Controlled	The user, not the computer, initiates and controls all actions. Make it clear to users what effect their actions have.

Provides Immediate User Feedback	The interface should keep the user informed and provide immediate feedback. • User feedback is primarily visual. For example, using a client side script, e.g., JavaScript, to change the color of links as the mouse passes over them gives the user immediate feedback (audio can also be used to provide feedback, but should be used selectively). • Immediate feedback creates the perception of high performance. • The resulting feedback should be appropriate to the task.
Direct	Users must see the visible cause-and-effect relationship between the actions they take and the objects on the screen. This allows users to feel that they are in charge of the computer's activities.
Builds on Familiar Concepts	Each web site should adopt elements of other common FSA web sites as possible in order to leverage the user's knowledge across the entire range of FSA internet applications. There should be a degree of cohesiveness across FSA sites. This may be accomplished by reusing layout, headers, and navigational images.
Consistent	A consistent interface allows users to apply previously learned knowledge to new tasks. Effective applications are both consistent within themselves and consistent with one another. Within a web site, fonts, colors, and images should be used consistently. Related applications and forms should work similarly.
Draws Eye Toward Important Areas	Avoid using bright colors or animation in a way that competes with content. Use color, fonts, images, and layout to focus attention on the key elements of the page, while maintaining compliance with Section 508 guidelines.
Avoids Redefining Commonly-used Behaviors of Elements	Do not redefine the behavior of elements the user knows from other contexts. For example, avoid using underline text for emphasis because users expect underlines in web pages to indicate links.
Forgiving	Users make mistakes. User actions should be reversible. A good interface facilitates exploration and trial and error learning. For transactions, submit for an action with severe consequences should prompt users for a confirmation (e.g., modifying user profile).
Breaks User Interactions into Small Chunks	Minimize the number of clicks necessary to access data or perform a function. For example, avoid organization structures within web sites that are more than three levels deep. Map the site structure logically in ways that match the user needs and tasks.

2.1 Requirements and Recommendations

This section presents items as either requirements or recommendations. A **Requirement** means that the specified item is an FSA standard. All GUI requirements must be followed in order for the web page or web site to be published. A "check" indicates that the specified item is a requirement, and the item usually contains the word "must."

For example:

√ Never use spaces in a file name. If a file name must be separated, use an underline to join the two words together (e.g. "file_name.html").

Recommendations are items that should be followed unless there is a sound business reason not to do so. Recommendations are indicated with triangles, and they usually use the word “should.”

For example:

Δ Developers should provide alternatives to audio or video, such as a textual transcript. Contact the Webmaster with questions about web style.

3 Web Content

3.1 Writing Style

Δ Online documents should be concise and structured to make it easy for web users to scan the page quickly. Most customers are looking for information, rather than reading a web page word for word.

Δ Keep headings, lists, and other elements that are intended to grab a user's attention clear and precise.

Δ Place important facts and conclusions at the beginning of a text where users and the search engine can find them quickly.

3.2 Spell Checking/Proofreading

√ All documents, as well as text on images, should be carefully and thoroughly checked for spelling or grammatical errors before posting them.

3.3 Information Priority

√ All web documents must be evaluated for importance and priority. Information and links of high importance and priority should have greater prominence on the page.

3.4 Security Classification

√ All web documents must be evaluated as Public, Internal, Confidential, or Privileged. Confidential and Privileged information must be clearly labeled. Developers may choose whether or not to label Public and Internal information. Refer to Table 2 for specific definitions of the foregoing classifications.

√ Security access to a web page should adhere to the most restrictive class of content on that page. For example, if a web page contains both Public and Confidential information, then developers should follow the confidential usage and access restrictions.

Table 2: Security Classifications

Classification	Description	Usage Restrictions
Public	Information can be classified as Public only after it has been formally released for public distribution by an authorized FSA person or channel. Public information requires no protection against disclosure; it can be freely communicated with everyone.	None
Internal	Information must be classified as Internal if it does not clearly belong in any other	None

	classification (Public, Confidential, or Privileged). Most day-to-day information sources are for FSA's internal use only and should not be communicated outside FSA.	
Confidential	Information must be classified as Confidential if it should not be disclosed to all employees; could damage FSA in some way if it were misinterpreted, or is considered private.	• Web pages that send or receive Confidential information must be encrypted, the identity of the recipient confirmed by FSA Security Team, and an audit log maintained and archived for a period of no less than six months. • Privileged information and documents must be clearly labeled with the "Confidential" image.
Privileged	Information must be classified as Privileged if FSA legal department or counsel created or directed the creation of information under attorney-client privilege. This classification has special protections under law that other classifications do not.	• Publishing Privileged information requires the written consent of the legal department. • Web pages that send or receive Privileged information must be encrypted, the identity of the recipient confirmed by FSA Security Team, and an audit log maintained and archived for a period of no less than six months. • Privileged information and documents must be clearly labeled with the "Privileged" image.

4 Page Structure and Appearance

Throughout the website site, developers should have consistency not only in the layout grid, but how to present the images, the style of the navigation bar, the fonts and functionality of web site. This makes the site more intuitive, visitors more comfortable and more likely to continue navigating the site. It lets the users know that they're still at the same site. However, still try and provide a difference between sections or sub-sites so the visitor can distinguish different areas in the site. This helps in navigation as it helps the visitor to know where they are in the site.

4.1 Typography

4.1.1 Alignment

- Text in the main body of a web page should be left aligned. Centering paragraphs of text on a page makes the content hard to read.
- Columns of numbers must be right justified. Generally this is easiest to do within TABLES. Numbers within a column must have the same number of decimal digits so they align on their decimal points.

4.1.2 Fonts

- Use typographic styles sparingly to ensure a clean, consistent flow to the content.

- No more than two fonts should be used on one page. There should be sufficient contrast between one font and another, and between text blocks, headlines, and the surrounding white space.

- Fonts used in a web site should come from the default collections that are installed with Macintosh or Microsoft Windows and from the Truetype fonts that Microsoft has made publicly available online at http://www.microsoft.com/opentype/.

- Font styles should be used consistently throughout the web site. Arial is generally recommended for legibility.

4.1.3 Heading Tags

- Use headings sparingly to ensure a clean, consistent flow to the content.

- There should be only one <H1> heading per page. The page name specified by the <H1> tag generally should be identical to the title specified in the <TITLE> tag. The H1 tag represents the main theme/idea/purpose of the page.

- A graphic with ALT text may act as a header. Other HTML headers used on the same page should take into account the relative size of the graphic and be sized accordingly. For example, a graphic banner might replace the top-level <H1> on a page; therefore, any HTML headers used below it would be <H2> and smaller.

- Lower-level headers (e.g., <H2>, <H3>, etc.) may be used if appropriate to the document.

- Documents which are divided into multiple pages should include the document title and publication date in italics above the top-level header, to help identify the document to users who may arrive at the page without knowing its context, (e.g., the result of a full-text search).

- Use <H4>, <H5>, and <H6> for captions, or when you have many levels of sub-headings. In general it's best not to go lower than <H4>.

- Header markup should not be used to emphasize entire paragraphs. Generally, section titles and other text marked with the Subhead style in a word-processing document should be marked up as lower-level HTML headers instead of simply appearing in bold.

- Cascading Style Sheets may be used for specifying the appearance of header tags.

- Additional text formatting tags (bold, italic, etc.) should not be applied to headers because they can confuse some browsers (like Lynx). No validity (syntax) checker will permit inappropriate use of heading tags because they are intended to convey the outline or structure of a document. Most browsers default to bold to display headers anyway.

4.1.4 Formatting

- No text should be underlined unless it is a link. This includes text that appears on graphics as well as typed text and headings. Be particularly aware of the need to make modifications to the use of underlined words and sentences when converting existing word processing documents to HTML.

- When creating a new document using a word processor that will be converted to ASCII or HTML, do not use the word processing program's style sheets to produce "all capitals" or other formatting effects. The effects will be lost during the conversion.

4.2 White Space

- Although the horizontal rule (<HR>) is a popular tool of HTML and makes it easy to create divisions between sections, avoid using horizontal rules if at all possible. The use of horizontal rules can make the screen look cluttered and detract from the overall design and readability of the page.

- In place of horizontal rules, use 'white space' to create divisions between sections. White space is a blank area between sections. This is easily achieved using a clear single pixel.

- Use <BR> when you need to break a line but leave no blank space. The <BR> tag is used similarly to a RETURN key in a word processing program.

- To skip a line, use the paragraph tag, <P>. However, a sequence of multiple <P> tags is treated as a single blank line. If multiple blank lines must be inserted, use the <PRE> and </PRE> tags with blank lines inserted between them.

4.3 Page Width

√ Use relative table widths and percentages when designing the width of a page. Do not design pages that are dependent on a fixed screen width or use absolute pixels. Screen width depends on the user's monitor size, browser and operating system.

√ As a widely used government agency, FSA web pages must meet the needs of as many users as possible. With that in mind, all FSA web pages must be created to accommodate users with low screen resolutions. This minimum level of screen resolution for web applications is a resolution of 800 pixels wide by 600 pixels high – the VGA standard resolution. Even though most graphic cards and monitors now support greater resolution, many users leave their machines configured at VGA resolution because of small (less than 17") screen size. It is easy to overlook this factor in a web development environment where more advanced machinery is on hand.

Δ At a minimum, ensure that your design degrades nicely to work with a minimum page width of around 470 pixels.

4.4 Page Length

√ Large or complex documents intended for online viewing, typically those larger than two screens in length, should be divided into multiple, smaller files with a menu page or Table of Contents describing the various sections. Page length should be based on the standard screen resolution for the FSA client platform. Two to three scrollable regions are generally acceptable (roughly equivalent to one printed page).

Δ If possible, files should be divided along logical break points such as chapters or sections.

Δ Pages longer than two screens should be broken up into smaller pages and hyperlinked from a Table of Contents.

Δ To assist users in navigating sectioned documents, each page should include standard navigation buttons at the bottom with links to the table of contents and previous and next section. The first and last sections of the document should omit the previous and next link respectively.

4.5 Color Usage

√ All pages must have a light colored or white background and dark colored text. The navigational framework within sites may be the one area that departs from this. Regardless, there should be adequate background / foreground contrast to ensure legibility.

√ Background and text colors must remain constant throughout any given web site.

Δ More than two or three colors of highlight will generally lead to a confusing interface. If that many colors are required to make your message known, you should consider giving the user the capability to limit the information they have to deal with at any given time.

- For the sake of those who might be colorblind, choose colors not only on the basis of base-color (Red vs. Green), but also on basis of contrasting brightness. Consider whether another interface device (use of an icon, for instance) could convey the same message. Also be aware that some color combinations do not work well together. Avoid saturated blue / red (text to background combos) – as they are difficult to read

4.5.1 Web-Safe Color Palette

- All non-JPEG graphics and color elements should use the 216-color cross-platform web palette. Use the palette to create all blocks of colors, text, lines, etc., and to reduce the color depth of images. Using the standard palette prevents images from dithering, or looking dotted and fuzzy, in the browser.

5 Multimedia Usage

5.1 Images

5.1.1 Standard Image Formats

√ Images should be saved in either the Graphic Interlaced Format (GIF), including GIF, GIF89, and animated GIFs, or Joint Photographic Experts Group (JPEG) format.

5.1.2 GIF vs. JPEG Format

Δ GIFs should be used for images that predominantly consist of solid colors, blocks of colors, lines, and/or text. The Graphics Interlaced Format reads images line by line from left to right. Therefore, the more color variations per line, the greater the size of the GIF file. GIFs use no more than 256 colors, of which only 216 are web-safe; try to use even fewer if at all possible (especially for buttons or simple graphics or icons).

Δ If an image has many color variations, and if attempts to "GIF" this file using the 216-color web palette results in substantial quality loss, use the JPEG format. The JPEG format is especially efficient for images made up of many colors, complex gradients, and photographs. All photographs (except small duotones or grayscales) should be compressed using the JPEG format. If space is constrained, reformatting an image in JPEG at the "Medium" setting would resolve this matter. If better quality is demanded, at the price of a higher file size, using JPEG at the "High" setting is a viable option.

√ GIF images must be saved as "Interlaced" instead of "Non-Interlaced." This makes the page appear to load faster for users with slower connection speeds. The image will be displayed roughly at first (via intermediate pixel lines) and will become more intelligible as the browser receives more data.

5.1.3 Image Attributes

√ All images must contain <ALT> text in case the image does not load in the browser, or in case it loads slowly. Including <ALT> text also ensures accessibility for users with text-only browsers, browsers with images turned off, and screen-readers.

√ Images must contain the WIDTH and HEIGHT attributes to help images load faster.

5.1.4 Image Maps

√ Image maps should have clearly defined "clickable" areas so users are not confused about what will happen when they click on an image map.

Δ Use client-side image maps rather than server-side image maps.

5.1.5 Location on Server

√ All top-level images are located in the common images folder of the root directory, and the sub level images are in their respective images folder.

√ All common graphics in a web site should be put in a common images directory (e.g. "/images") and should be made available for all FSA developers to use.

5.1.6 Consistency

√ The style of navigational buttons should be consistent throughout sections of a web site or application.

√ The fonts used for text and on navigational graphics and graphical buttons should be consistent throughout the site and should be easy to read. Arial is recommended as a legible font that will work across browsers.

5.2 Audio and Video

5.2.1 Audio and Video Formats

√ Use cross-platform file formats, such as MP3, RM and AU for non-streaming audio, and MOV or MPEG for non-streaming video.

5.2.2 Audio and Video Attributes

√ All audio or video components must include ALT tags for text-only browsers and screen-readers.

5.2.3 Accessibility Issues

√ Should provide alternatives to audio or video content, such as a text-based transcript.

5.3 Plugins

5.3.1 Acceptable Formats

√ The following plugins are acceptable for use on FSA web sites:

- Macromedia Flash
- REAL Video and Audio
- Acrobat Reader for PDFs

5.3.2 Accessibility Issues

√ Must provide alternatives to files requiring plugins, such as a text-based transcript.

6 Web Page Templates

Consistent and an easy-to-navigate layout is essential to good web page design. Use the sections below as guidelines for developing clean, simple, and accessible page design.

Web site design templates are provided for developers to use as a general guideline when creating web pages. Developers should use the general layout approach detailed in this section to arrange the content on a web page. These templates contain information about standard areas, headers, and footers that are expected elements on every FSA web page.

6.1 Home Pages

The home page is the first page of a web site that users see. The home page is often a launching pad for other sections of the web site.

6.1.1 Menu-Based Home Pages

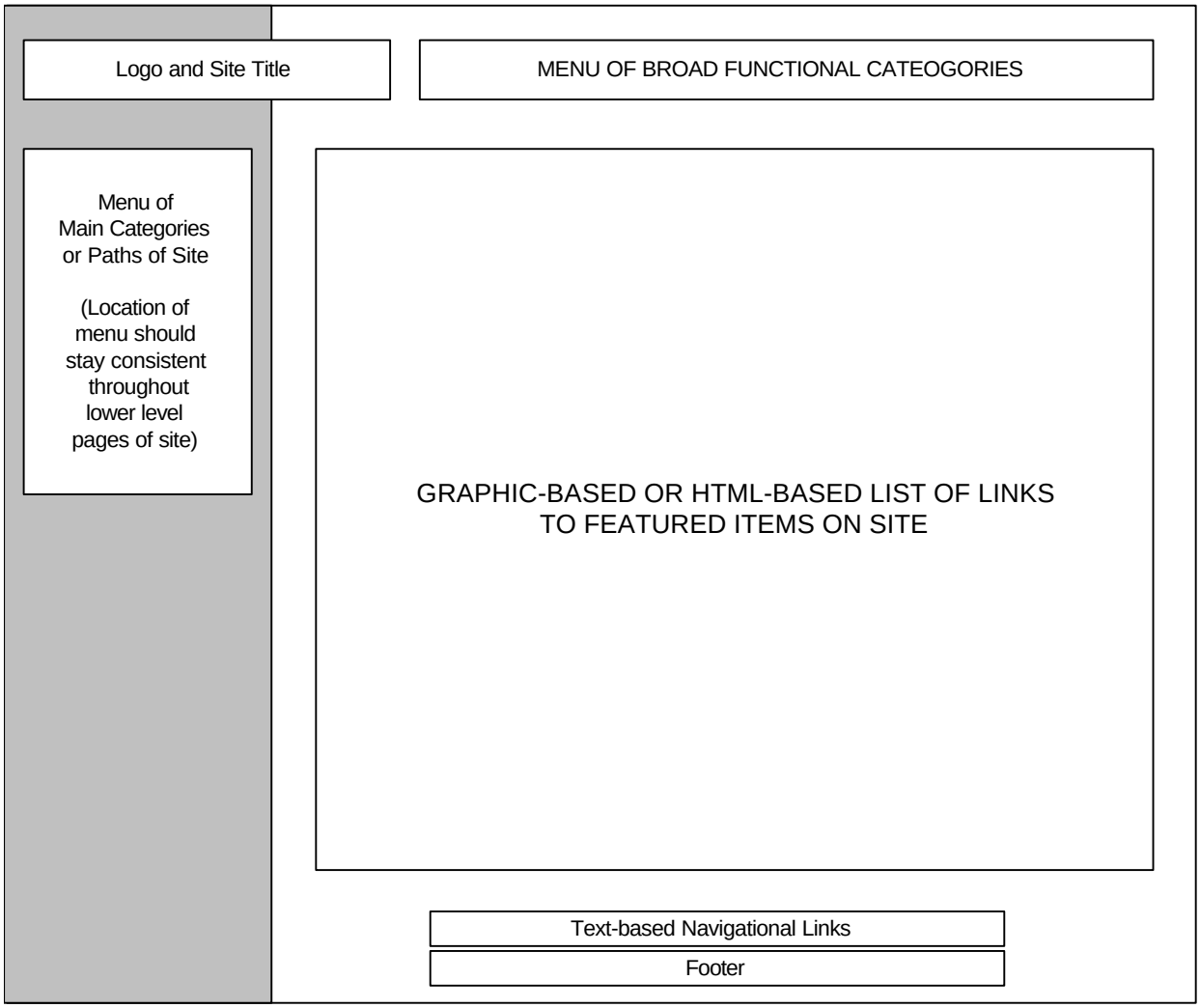
The home page should impart, graphically and informationally, that it is entry point of the site. Menu-based home pages (Figure 1) that offer lists of links and broad paths to the rest of the site are the most common type of home page. Larger web sites that are doorways for several audiences fare better with this type of home page design. The application's home page (http://www.ed.gov/) uses this type of home page design.

Menu-style home pages typically use either text-based HTML links or graphic image maps, or a combination of both. The menu style makes it possible to include a large number of site paths in a small amount of page real estate.

Category and path names should be strategically chosen and should be clear which audience or interest groups they serve, or should define areas of the site that contain specific information or achieve certain goals.

Links back to the home page should be available from every page in the site. The home page should promote a distinct design (while introducing common navigational and design elements) from the other pages. The overall organization and structure of the site should be conveyed by home page design. It should also clearly introduce the site's purpose and be segmented by what the key modules or sections and functions of the site are. There should be prominence (in terms of placement) given to the highly important, frequently accessed functions, modules or information, with consideration to audience's need.

Figure 1: Sample Layout for Menu-Based home pages



6.1.2 Portal Design Home Pages

The best-designed portal sites allow users to find the topics or information that they are looking for, and easily print or download what they find. (Note: not all portals are for download or print)

Graphics should be minimal, and content and menu structure must be carefully organized to support fast search and retrieval, easy downloading of files, and convenient printing options.

User contact time is typically brief on portal sites, particularly on the home page. Shorter "eyeball" time is better, as it implies that users are finding what they are looking for quickly.

The screenshot below is of the FSA Students Portal. The elements discussed in this section are represented in Figure 2, for example, the menu options are mirrored in the central portion of the

interface (road signs) as well as the tabs above the graphic pane. The focal point of the interface is the 'road sign' graphic which also links to the web site's content.

Figure 2: Sample Layout for Portal Design home pages

Section 6.3.1 – Standard Header

6.1.3 News-Based Home Pages

Web sites that are repositories of news-based information typically organize their home page into rectangular chunks of information (e.g., headlines and one or two sentences of block text) in order to mimic the design of a newspaper (Figure 3). Sometimes this design pattern is incorporated into portals.

These types of web sites generally offer users access to quickly changing news, calendar events, and message alerts. Particular care should be given to keep the content on these sites extremely fresh, such as changing the content hourly, daily, or at most, weekly.

While the content itself should remain fresh, choosing a standard framework for organizing information on the home page is important. The overall layout of the home page should remain consistent, or users are likely to become confused on return visits.

Figure 3: Sample Layout for News-Based home pages

6.1.4 Lower-level Pages

The second and third level (or tier) pages of every web site should use a standard layout grid that all internal pages of the site will share. As such, it is very important to design a strong, logical page grid that may be used consistently throughout a site for menu pages versus content pages.

It is important to establish a regular, repeating pattern of carefully organized pages of text and graphics to help the user establish the location and organization of information and increase legibility.

Templates for establishing layout of lower-level pages work consistently well for the following types of functional pages:

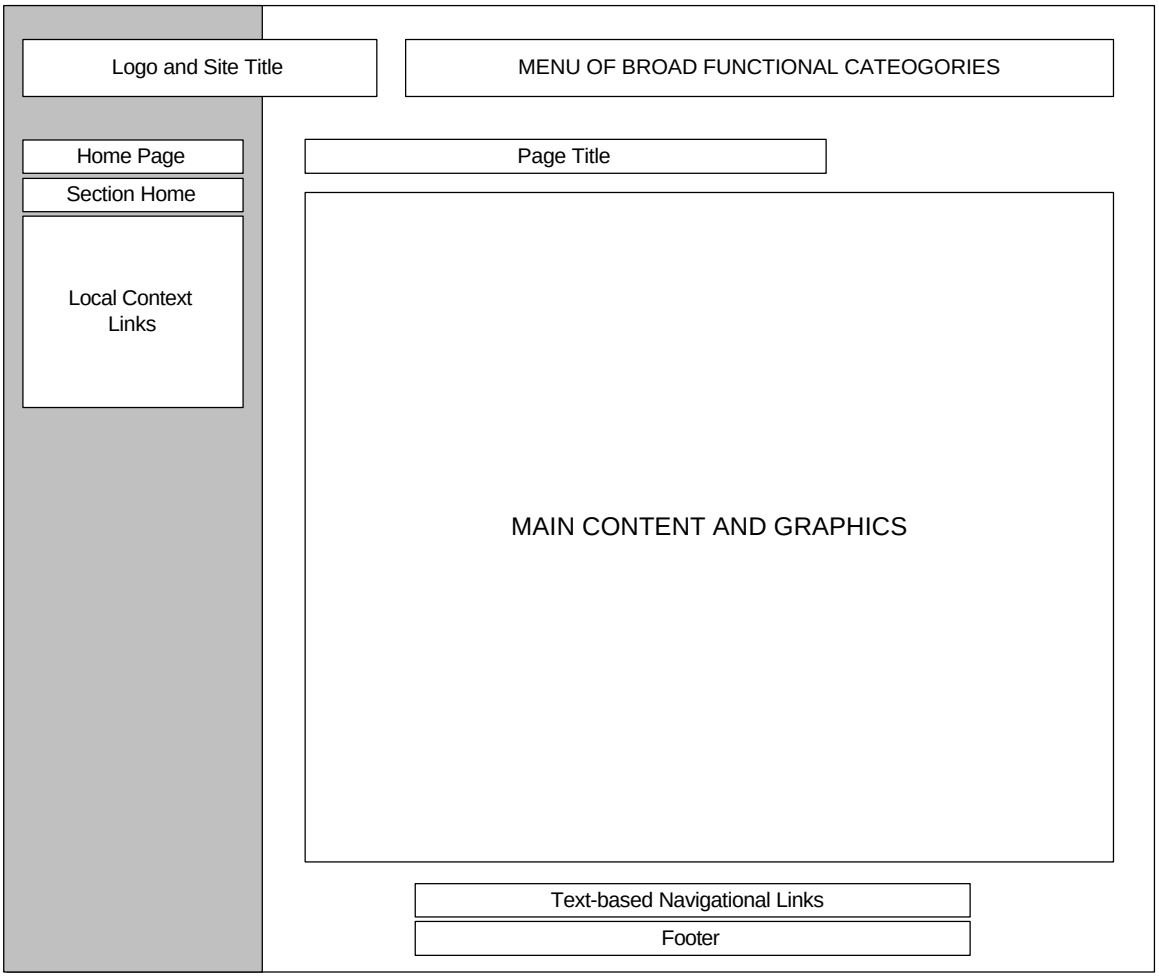
Second and third level pages (except for section or sub-site home pages, which typically reflect the design template established by the web site's home page)

- *Content-oriented pages* (web pages that are text-heavy in the body of the page)

- *Form-based data-entry pages* (web pages whose primary purpose is to collect information from the user, typically through the use of web-based forms)

Generally, it is recommended the same approach to organize information on the page for all of the above styles of pages, is followed. Often lower-level pages are content-oriented or form-based pages.

Figure 4: Sample Layout for Second-level Pages



6.2 Standard Broad Functional Categories

Functional pages described in this section, such as the site map and search page, are consistently found in web sites and the expectation is that they be included in all web sites where appropriate.

Pages that fall under this category are usually included in the web site's main functional navigation bar, which is located at the top of the web page in the menu of broad functional categories.

6.2.1 Site Maps and Table of Contents

Each web site or web application is expected to maintain a Site Map that shows the structure of the site. Preferably, the Site Map would list the main content areas of the site and hyperlink from the Site Map to these areas, which will ensure quick and easy navigation from the Site Map to the main areas of the site.

An extensive listing or graphic overview of internal site links offers users quick access to content. Search facilities are not a good substitute for a clear, well-organized list of a site's contents -- such facilities should supplement the site's content.

Text-based tables of contents or indexes are generally more efficient and informative than a graphical metaphor, such as a hierarchical branching diagram. Furthermore, graphical site maps are more difficult to maintain, such as when a site is reorganized or new information is added or deleted.

Modifications to the Site Map should be coordinated through the primary web manager or web developer.

6.2.2 "What's New" Pages

In the first few years of the World Wide web, the "what's new" page was used extensively as a repository for all changes to content that had been made during a certain time frame.

However, most web sites are now continually updated and refreshed to keep content "fresh." In order to inform users about new content, the developers could place a "new" graphic next to each updated item.

Sites that are complex, with many levels of information spread over a great number of pages may prefer to maintain a "what's new" page, rather than posting "new" graphics everywhere.

6.2.3 Search Features

All FSA sites should offer a robust (within site) search facility for users. Keyword searches allow the user to search for specific information that they know exists but are not sure where it may be located in the hierarchy of the web site, or if the information they are looking for is not listed in the site's table of contents or site map.

Indicate whether the search engine searches the entire web site, a collection of documents (such as a subsection), or a database. Sites that are particularly complex should offer readers a menu that allows them to limit their search to a specific area.

6.2.4 Contact Information and User Feedback

Provide links to contact information, in a prominent contact page in each site. The following information is typically provided on a contact page.

- Important Email addresses
- Street addresses

- Technical support phone numbers
- Organizational phone numbers
- Fax numbers
- Maps, travel directions, or parking information

The web site's Webmaster email address, used for technical web site support, should be posted in the footer of every web page (see Section 6.3.2 - Standard Footers).

Provide a web form to collect user information and feedback. A database may be used to store and analyze user input. It is advisable to have a web site customer service infrastructure in place to handle incoming questions from users.

6.2.5 FAQ Pages

Frequently asked questions pages offer a central location for the most commonly asked questions received from users.

FAQ pages are primarily used to provide self-service support and information and are intended to improve users' understanding of the information and services offered while reducing the burden on operations support staff.

6.3 Standard Headers and Footers

6.3.1 Standard Headers

The header is a set of information placed at the top of every web page (Figure 2). Every document should contain the following information in a standard header:

- Navigation tags
- Privacy Information
- Security Information
- Notices

A virtual include file may be used and referenced in the document to hold the header information. For example:

```
<!--#include virtual="/includes/header.html"-->
```

Sample code for a standard header follows:

```
<!-- HEAD - PRIVACY, SECURITY, NOTICES -->
  <tr>
    <td width="100%" align="right" valign="bottom" class="sepline" colspan="2">
    <a href="#content">Skip Navigation</a>

     | <a href="/PORTALSWebApp/fp/privacy.jsp">Privacy</a>
     | <a href="/PORTALSWebApp/fp/security.jsp">Security</a>
     | <a href="/PORTALSWebApp/fp/notices.jsp">Notices</a></td>
          <td><img src="/fp/images/1.gif" width="3" height="0" border="0" alt=""></td>
   </tr>
```

6.3.2 Standard Footer

- The footer holds information placed at the base of every page. Every document will contain the following information in a standard footer, separated from the body of the page by an appropriate amount of white space (usually two <BR> tags):

- Link back to the top of the page (can be omitted if the page fits onto one screen)
- Link to an appropriate home page (for the document, collection, project, program, organization, or Department) with the assumption that additional pages will also link back to home and other pertinent points within the site.
- Links to additional documents in same section
- Date that the document was last modified
- Responsible engineer / contact person with person's name, email address (and mailto: link)

- A virtual include file may be used and referenced in the document to hold the footer information. For example:

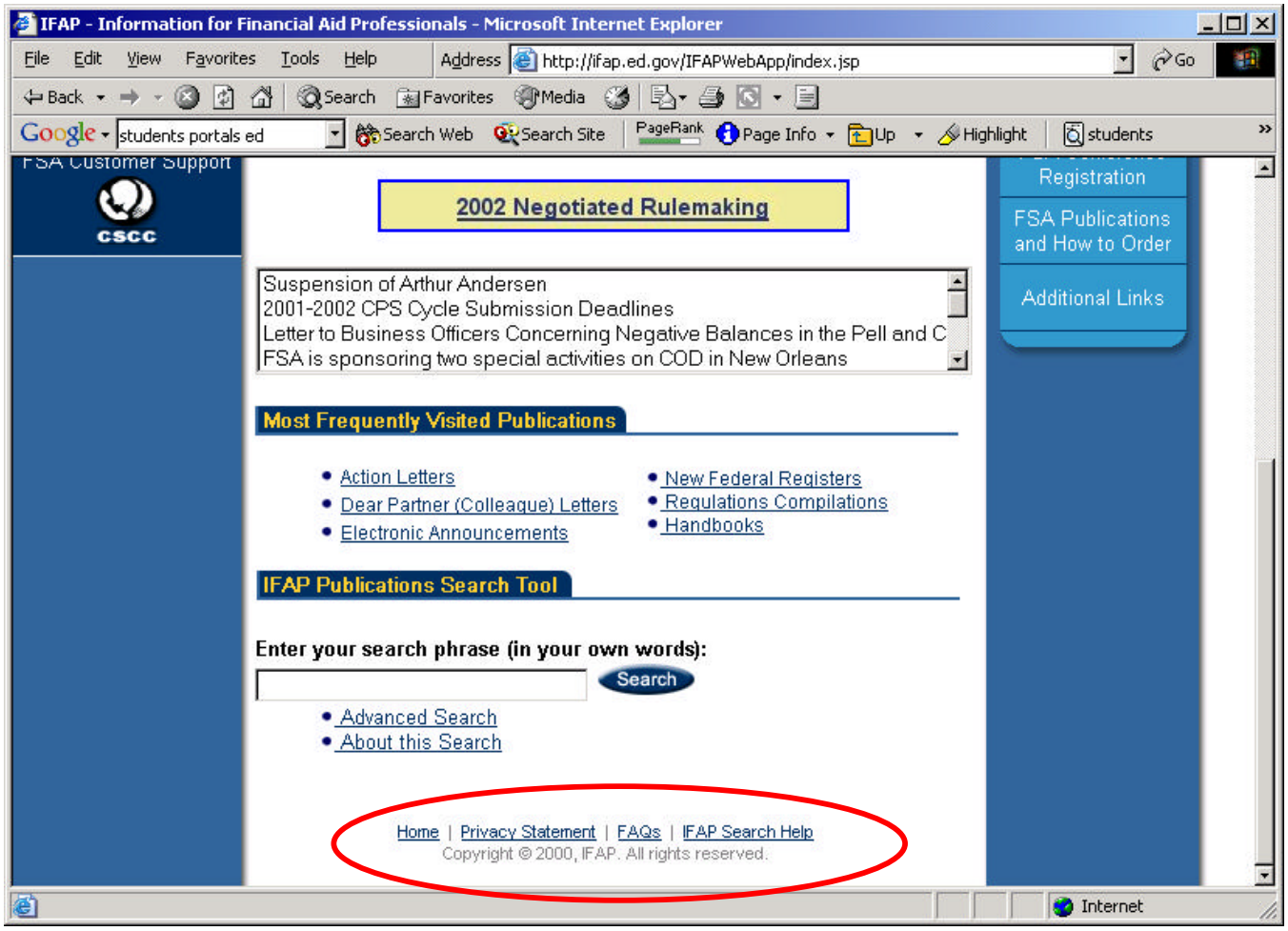
<!--#include virtual="/includes/footer.html"-->

- Sample code for a standard footer follows:

Client (***note this is included at the beginning of every page already***

```
<!-- FOOT -->
<table width="100%" cellspacing="0" cellpadding="0" border="0">
        <tr>
                <td width="800" align="right" class="sepline">
                        <br><br>
                        <a href="/PORTALSWebApp/fp/index.jsp">
                        <b>Home
                        </b>
                        </a> | <a href="/PORTALSWebApp/fp/aboutus.jsp">About Us</a>
                        | <a href="/PORTALSWebApp/fp/sitemap.jsp">Site Map</a>
                        | <a href="/PORTALSWebApp/fp/feedback/survey.jsp">Survey</a>
                        | <a href="/PORTALSWebApp/fp/help.jsp">Technical Help</a></td>
        </tr>
        <tr>
                <td width="100%"><hr width="100%" size="1" color="#000063"></td>
        </tr>
</table>
```

Figure 5: Sample Layout for Standard Footer



6.4 Standard Style Sheets

A cascading style sheet may be used to apply all attributes to the page, and is included within the <HEAD> tags at the top of every HTML document, this allows for changes to be made to the entire site quickly:

```
<LINK REL=STYLESHEET HREF="/global.css" TYPE="text/css">
```

6.5 Custom Server Error Pages

The most commonly served error page is the widely known “404 File Not Found” page. Usage of custom-designed, and standard set of error pages that are consistent with the web site’s look and feel is recommended. The quality of interaction for users is improved, particularly for web applications that offer useful functionality, such as explain the error, suggest alternatives, or provide links to home page, site map, or search page.

7 HTML Coding Standards

7.1 Major HTML Tags

√ Standard HTML tags are required for correct and consistent document presentation. The complete structure of a standard HTML 4.0 document is provided in Table 3 below.

Table 3: Standard HTML document

Tag Function	Use of standard HTML tags within an HTML document
Document Type	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
Opening HTML tag	<HTML>
Opening HEAD tag	<HEAD>
Title	<TITLE>Brief but Descriptive (and Unique) Title of Page</TITLE>
Closing HEAD tag	</HEAD>
Opening BODY tag *The body tag contains the content displayed by your document.*	<BODY BGCOLOR="#ffffff" LEFTMARGIN="0" TOPMARGIN="0" MARGINHEIGHT="0" MARGINWIDTH="0" ALINK="#333399" VLINK="#333399" LINK="#333399" BACKGROUND="../images/background.gif"> Document Content.
Closing BODY tag	</BODY>
Closing HTML tag	</HTML>

7.1.1 Document Type

√ All FSA pages must support the HTML 4.0 standard. Although many web pages do not include a reference to document type, you must include the following as the first line of the HTML document so that all FSA pages are identified as HTML 4.0 based:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
```

7.1.2 Meta Tags

√ All pages must contain the <META> tags. An example of a standard <META> tags is provided below, but others are acceptable

```
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
```

7.1.3 Titles

√ Each page must contain a descriptive title (in the <TITLE> tag). This is the title that will appear when people search for information, when they create a “Favorites” listing in their

browser, and when they print the page.

Δ The title should be as short as possible but should be fully informative and specific (e.g., " Client - Site Guide " is preferable to "Site Guide"). Although the title is often overlooked because it does not appear in the body of the document, it is important because it is frequently used to identify the document on hotlists, search result sets, and site indexes.

Δ The TITLE must provide enough contextual information to be used as a useful bookmark title. Do not assume that the user will be aware of the page's context, since the user might browse to the page directly from an external page or from a different section of the web site. The TITLE must not exceed 60 characters to avoid clipping when it is displayed in the title bar of certain browsers.

√ The <TITLE> tag is inserted between the opening and closing <HEAD> tags. The title will appear at the top of the respective browser and is also used by the browser to "bookmark" the page for future access. Additionally, some search engines use the <TITLE> as primary indexing point. The title tag is closed with </TITLE>.

```
<HEAD>
<TITLE>Client - Site Guide</TITLE>
</HEAD>
```

7.2 Readability of HTML Code

7.2.1 Use of Case in HTML Tags

√ HTML is not case sensitive, but do not mix case with tags. **Use lowercase for all tags, names, and attributes.** The only exception is special characters. (Note that newer standards that are coming in the future prohibit use of uppercase).

7.2.2 Use of Comment Tags

Δ The use of HTML comments to identify major and minor sections of code, components, functions, and navigational areas, and to identify the purpose, use, and outcomes of functions and components, is highly recommended (e.g. **<!-- This is a comment -->**).

Δ Refrain from using comments to "comment out" a section containing other HTML tags because some browsers have trouble processing these areas, and unexpected errors may result.

7.2.3 Line Length

√ Do not create lines longer than **160 characters**. Although line length is unimportant to browsers, many editors handle the wrapping poorly making it hard to edit the HTML files.

7.2.4 Use of White Space

Δ Browsers ignore white space between lines. On a given line, multiple blank spaces are treated as one space and tabs are ignored. (None of the above apply between <PRE> and <PRE>, however). Take advantage of white space by structuring and properly indenting your

code so that your HTML code is more readable. Bear in mind that tabs and carriage returns can add significant weight to file size. Use of indentation is discouraged.

7.2.5 Use of Double Quotes in Attribute Values

Although not all browsers require this, you should place double quotes around all the values of an attribute, not just when the value contains embedded spaces. (Technically, double quotes are only needed if you have a non-alphanumeric in the string, but it's best to always use them).

Note: Future standards will require double quotes on all names and attributes.

```
<INPUT TYPE="submit" VALUE="Submit">
```

7.3 Frames Usage

Frames are not to be used on FSA web sites. Frames interfere with accessibility for persons with disabilities. Instead of using frames to organize content, use tables or INCLUDE files. Macromedia Dreamweaver, a standard tool for developing static HTML pages, is an easy-to-use HTML editor that allows users to use tables for organizing layout.

8 Accessibility Standards

8.1 Purpose

The FSA considers accessibility to information a priority for all employees and external customers, including individuals with disabilities. FSA is issuing accessibility standards for electronic and information technology covered by section 508 of the Rehabilitation Act Amendments of 1998. Section 508 requires that when Federal agencies develop, procure, maintain, or use electronic and information technology, they shall ensure that the electronic and information technology allows Federal employees with disabilities to have access to and use of information and data that is comparable to the access to and use of information and data by Federal employees who are not individuals with disabilities, unless an undue burden would be imposed on the agency. Section 508 also requires that individuals with disabilities, who are members of the public seeking information or services from a Federal agency, have access to and use of information and data that is comparable to that provided to the public who are not individuals with disabilities, unless an undue burden would be imposed on the agency. Section 508 establishes state government also, by extension through the Assistive Technology Act of 1998, to procure information technology that is accessible.

While a product that meets these requirements ensures minimum accessibility for individuals with disabilities, the FSA encourages software and technology developers to be creative and maximize their design of software that is universally accessible. More specific recommendations for how to design universally accessible software can be obtained from links on the following web site:
http://www.w3.org/WAI/

8.2 Section 508 Guidelines

The following are the specific guidelines for web Based Intranet and Internet Application and Operating System Guidelines based on Section 508:

8.2.1 Keyboard Access

- The software program must provide keyboard access to all functions of the application. All actions required or available by the program must be available with keystrokes.

- Clear and precise instructions for the use of all keyboard functions shall be provided as part of the user documentation.

- The software must have a logical tabbing order among fields, text boxes and focal points.

- There must be a well-defined, programmatically exposed focal point that moves as the input focus changes so that assistive technology can track the focus.

- The software shall not interfere with existing accessibility features built into the operating system, such as Sticky keys, Slow Keys, Repeat Keys, high contrast mode in Microsoft Windows.

Δ Avoid using timed responses if possible. If used, the ability to modify the timing parameter, by individual user, is necessary.

8.2.2 Icons

√ All icons and images shall have clear precise text labels, <ALT> tags or tool tips included on the focus or provide a user-selected option of text-only buttons.

√ The use of icons shall be consistent throughout the application.

√ Provide pull-down menu equivalents for Icon functions (menu, tool and format bar).

8.2.3 Objects (Controls)

√ Painted text is not accessible to all users. Use system text drawing tools so that screen reader software can interpret the text. The minimum information that must be provided to an assistive technology is text content, text input caret location and text attributes.

√ Sufficient information must be provided about user interface object so that assistive technology can identify and communicate how the objects are used (e.g., the object is an edit field or a check box which is checked).

√ Redundant text links shall be provided for each active region of a server-side image map.

√ Client-side image maps shall be provided instead of server-side image maps except where the regions cannot be defined with an available geometric shape.

√ Row and column headers shall be identified for data tables.

√ Markup shall be used to associate data cells and header cells for data tables that have two or more logical levels of row or column headers.

√ When electronic forms are designed to be completed on-line, the form shall allow people using assistive technology to access the information, field elements, and functionality required for completion and submission of the form, including all directions and cues.

√ A method shall be provided that permits users to skip repetitive navigation links.

√ Software shall not use flashing or blinking text, objects, or other elements having a flash or blink frequency greater than 2 Hz and lower than 55 Hz.

√ Applications shall not disrupt or disable activated features of other products that are identified as accessibility features, where those features are developed and documented according to industry standards. Applications also shall not disrupt or disable activated features of any operating system that are identified as accessibility features where the

application programming interface for those accessibility features has been documented by the manufacturer of the operating system and is available to the product developer.

√ When electronic forms are used, the form shall allow people using assistive technology to access the information, field elements, and functionality required for completion and submission of the form, including all directions and cues.

8.2.4 Sounds

√ Support the Sounds feature where built into the operating system, such as Microsoft Windows show sounds feature.

√ Provide an option to display a visual cue for all audio alerts. Minimize use of audio alerts in general.

√ Provide an option to display all audio information in text format, either as closed captioning, a pop-up window or other means in parallel with the audio content.

√ Provide an option to display all video information in text format, either as closed captioning, multi-media audio, a pop-up window, or other means in parallel with the video content.

8.2.5 Display

√ Do not use Color-coding as the only means of conveying information or indicating an action. Always provide an alternative or parallel method that can be used by individuals who do not possess the ability to identify colors.

√ The application must support user-defined color and high contrast settings for all user interface controls and client area content.

√ Do not use patterned backgrounds behind text or important graphics or provide an option to turn off background patterns.

√ Application colors should not be hard-coded, so that individual users can override application default fonts through the operating system, so printing and text displays can be changed.

√ Allow user adjustment of, or allow user to disable flashing, rotating or moving displays to the extent that it does not interfere with the purpose of the application. Minimize use of moving display elements.

√ Web pages shall be designed so that all information conveyed with color is also available without color, for example from context or markup.

√ Pages shall be designed to avoid causing the screen to flicker with a frequency greater than 2 Hz and lower than 55 Hz.

√ When bitmap images are used to identify controls, status indicators, or other programmatic elements, the meaning assigned to those images shall be consistent throughout an application's performance.

√ Textual information shall be provided through operating system functions for displaying text. The minimum information that shall be made available is text content, text input caret location, and text attributes.

√ A well defined on-screen indication of the current focus shall be provided that moves among interactive interface elements as the input focus changes. The focus shall be programmatically exposed so that assistive technology can track focus and focus changes.

8.2.6 Field Labeling

√ Position the descriptions or labels for data fields immediately next to the field, so that it is easy for screen reading software, used by individuals that are blind, to associate the labels with the corresponding fields. The preferred position would be flush against the right side of the field. Associate labels clearly and programmatically with controls and objects.

8.2.7 Documentation

√ Provide all manuals and documentation in electronic format as an ASCII text or HTML file. This should include text descriptions of any charts and/or graphs or pictures or graphics of any nature. This is done to ensure that the information presented in charts or graphs is available to screen readers and/or in Braille versions of the text.

√ Any reports, forms or documents that the application generates must be available in a "print to ASCII file" format.

√ Documents shall be organized so they are readable without requiring an associated style sheet.

√ Applications shall not override user selected contrast and color selections and other individual display attributes.

√ Color-coding shall not be used as the only means of conveying information, indicating an action, prompting a response, or distinguishing a visual element.

9 Additional Resources

This guide reflects the experience of many developers as well as documented “best practices” at many companies.

9.1 Style and User Interface Resources

Developers who are interested in additional information about style and other user interface issues should refer to the Internet resources listed in Table 4 below.

Table 4: Style and User Interface Resources

Style and User Interface Resources on the World Wide Web
Apple Web Design Guide http://applenet.apple.com/hi/Web/Web.html
Dave Siegel’s Tips for Web Writers and Designers http://www.dsiegel.com/tips/
Designing HTML Pages to Increase Accessibility to Users With Disabilities http://www.trace.wisc.edu/HTMLgide/htmlgide.html
Gif89a http://www.cnet.com/Content/Features/Techno/Gif89
Graphics Research Lab Notes on Writing for Interactive Media http://www.electric-pages.com/notes/n1.htm
Matterform Media Qbullets http://www.matterform.com/qbullets/index.html
National Center for Supercomputing Applications (NCSA) Review of Style Guides http://www.ncsa.uiuc.edu/edu/trg/styleguide/index.html
Style Guide http://www.cl.cam.ac.uk/users/gdr11/style-guide.html
Sun Guide to Web Style http://www.sun.com/styleguide/
The Intranet Journal Design Tools http://www.Intranetjournal.com/design.html

Usability Testing of WWW Designs
http://www.sun.com:80/sun-on-net/uidesign/usabilitytest.html

Web Pages That Suck
http://www.Webpagesthatsuck.com

Webmaster Magazine Writing Style Resources
http://www.cio.com/WebMaster/style.html

What is Good Hypertext Writing
http://kbs.cs.tu-berlin.de/~jutta/ht/writing.html

World Wide Web Consortium (W3C) Style Guide for Online Hypertext
http://www.w3.org/pub/WWW/Provider/Style/Overview.html

Yale University School of Medicine Center for Advanced Instructional Media Style Manual

- **First Edition:**
  http://info.med.yale.edu/caim/StyleManual_Top.HTML
- **Second Edition (Requires JavaScript):**
  http://info.med.yale.edu/caim/manual/contents.html

9.2 Accessibility Resources

Developers who are interested in additional information about style and other user interface issues should refer to the Internet resources listed in Table 5 below.

Table 5: Accessibility Resources

Accessibility Resources on the World Wide Web
Requirements for accessible software design http://gcs.ed.gov/coninfo/clibrary/software.htm **Rehabilitation Act of 1973** http://trace.wisc.edu/gofr_web/rehabact.html **Section 508:** Electronic and information technology accessibility guidelines (updated with 1998 Amendments) http://www.itpolicy.gsa.gov/cita/section508.htm **Federal IT Accessibility Initiative (FITAI)** Web Accessibility Training Information and Registration http://www.itpolicy.gsa.gov/cita/training.html

W3C Web Content Accessibility Guidelines 1.0
http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505/

Curriculum for Web Content Accessibility Guidelines 1.0
http://www.w3.org/WAI/wcag-curric/

Center for IT Accommodation (CITA)
http://www.itpolicy.gsa.gov/cita/

10 References

SCO Visual TCL Programmer's Guide and Reference
Appendix D, User interface style conventions
http://compy.ww.tu-berlin.de/VTcl_DOC/VTCLG/vtclgN.style_goodui.html

Web Style Guide
Patrick J. Lynch and Sarah Horton
Yale University Press
http://info.med.yale.edu/caim/manual/